

优化 TLV 编码规则*)

王沁 许娜 张燕 张晓彤
(北京科技大学计算机系 北京 100083)

摘要 抽象语法标记 ASN.1 是一种 ISO/ITU-T 标准,描述了一种对数据进行表示、编码、传输和解码的数据格式。ISO 协议体系中的应用层协议使用了 ASN.1 来描述它们所传输的协议数据单元。ASN.1 取得成功的一个主要原因是它与几个标准化编码规则(如基本编码规则 BER)相关。TLV 编码是指先对 Tag 编码,再对 Length 编码,最后对 Value 编码。BER 编码确定的编码方式就是这样的。在实际的通信中,通常特定类型编码中数据的长度是已知的,在数据字典中有明确的标识。提出了一种优化 TLV 编码的方法,使用一个比特的标志位,将长度字段作为编码中一个可选择的项目,而不是必须存在的项目,以此来缩短 TLV 编码的长度。在满足需求的前提下,优化后的 TLV 编码可以大幅度地提高数据传输效率。

关键词 ASN.1, BER, TLV, 优化编码

TLV Coding Optimization Rule

WANG Qin XU Na ZHANG Yan ZHANG Xiao-tong

(School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China)

Abstract ASN.1 is one of ISO/ITU-T standards, which describes a kind of data structure used in representation, encoding, translating and decoding. Application layer of the ISO protocol sockets use ASN.1 to describe protocol data unit used in the translation. The main reason of the success of ASN.1 is that it is closely related with several standard coding rules, such as BER. TLV code is composed of tag, length and value. This is the coding rule confirmed by BER. In actual communication, generally, the length of special type field is specific and unchanged, which can be definitely found out in data dictionary. Presented a method of optimized TLV coding method. By adding one-bit symbol, the method makes the length field become optional to shorten the total length of the data packet. This optimized method not only can satisfy the requirement of a variety of protocols, but also can greatly improve the translation efficiency.

Keywords ASN.1, BER, TLV, Optimization rule

1 引言

抽象语法标记^[1](ASN.1)是一种 ISO/ITU-T 标准,描述了一种对数据进行表示、编码、传输和解码的数据格式。它提供了一整套正规的格式用于描述对象的结构,而不管语言上如何执行以及这些数据的具体指代,也不用关心到底是什么样的应用程序:不论是非常复杂的还是非常简单的。在任何需要以数字方式发送信息的地方,ASN.1 都可以发送各种形式的信息(音频、视频、数据等)。ASN.1 和特定的 ASN.1 编码规则推进了结构化数据的传输,尤其是网络中应用程序之间的结构化数据传输,它以一种独立于计算机架构和语言的方式来描述数据结构。ISO 协议族中的应用层协议使用了 ASN.1 来描述它们所传输的协议数据单元(PDU),这些协议包括:用于传输电子邮件的 X.400、用于目录服务的 X.500、用于 VoIP 的 H.323 和 SNMP。ASN.1 的应用还可以扩展到通用移动通信系统(UMTS)中的接入和非接入层。

ASN.1 取得成功的一个主要原因是它与几个标准化编码规则,如基本编码规则(BER)-X.209、规范编码规则(CER)、识别名编码规则(DER)、压缩编码规则(PER)和 XER 编码规则(XER)。这些编码规则描述了如何对 ASN.1 中定义的数值进行编码,以便于传输,而不管计算机、编程

语言或它在应用程序中如何表示等因素。ASN.1 的编码方法比许多与之相竞争的标记系统更先进,它支持可扩展信息快速可靠地传输。在无线宽带中,这是一种优势。1984 年,ASN.1 就已经成为了一种国际标准,它的编码规则已经成熟并在可靠性和兼容性方面拥有更丰富的历程。

本文针对基本编码规则中涉及到的 TLV 编码规则进行了优化。通过将 TLV 编码规则要求必须出现的长度字段作为可选字段来缩短编码数据的长度,提高数据的传输效率。

本文结构如下:第 2 节论述 TLV 编码的定义和 DOCSIS 规范中的 TLV 编码;第 3 节分析 TLV 编码的特点,并提出优化 TLV 编码的方法;第 4 节阐述优化后的数据包长度该如何计算以及性能分析;最后是结论。

2 TLV 编码的定义和 DOCSIS 规范中的 TLV 编码

2.1 TLV 编码的定义

BER 编码有两种方式:一种是长度确定的编码方式,由 3 部分组成:Identifier octets, Length octets 和 Contents octets (可以和 TLV 对应);另一种是长度不确定的编码方式,由 4 部分组成,Identifier octets, Length octets, Contents octets 和 End-of-contents octets。每种类型都能够编码成长度确定的编码方式,但是有的类型不能够编码成长度不确定的编码方式

*)北京市科技重大项目“交互式数字电视信道传输核心技术开发”京科技发[2002]188 号。王沁 教授,博士生导师,博士,CCF 会员;许娜 博士研究生;张燕 博士研究生;张晓彤 副教授,博士。

式。标签(Tag)字段是关于标签和编码格式的信息;长度(Length)字段定义数值的长度;内容(Value)字段表示实际的数值。因此,一个编码值又称 TLV(Tag, Length, Value)三元组。编码可以是基本型或结构型。如果它表示一个简单类型的、完整的显式值,那么编码就是基本型(Primitive);如果它表示的值具有嵌套结构,那么编码就是结构型(Constructed)。

2.2 DOCSIS 规范中的上行通道描述符

DOCSIS 规范^[2]提出了针对网络特点的 MAC 协议,其基本思想是:将上行信道模型化为一连串的微时隙(Mini-slot)流,每个微时隙作为一个基本传输单元。这些微时隙按照传输方式分为两类:竞争传输和预留传输。用于竞争传输的微时隙称为竞争微时隙,它为所有 CM^[2]提供上行传输机会,因而易发生冲突。用于预留传输的微时隙称为预留微时隙,它专门为某一特定的 CM 提供上行传输机会。用预留方式传输的数据不会发生冲突。CMTS^[2]必须在一个周期性的时间间隔中发送一个上行通道描述符^[2](UCD),以定义上行通道的特征,必须为每个被激活的上行发送一条独立的报文。为了使通道的报文参数具有灵活性,把它们以类型/长度/数值(TLV)的形式进行编码,其中类型和长度字段各为一个八位组。在 DOCSIS 规范中,TLV 中的 T 被解释为类型(Type),本文认为这与 BER 编码中将 T 定义为标签(Tag)字段并不冲突。在这里,类型名称就是一种标签,用来指明后面的数值表示的含义。上行通道描述符的顶层结构如图 1 所示。

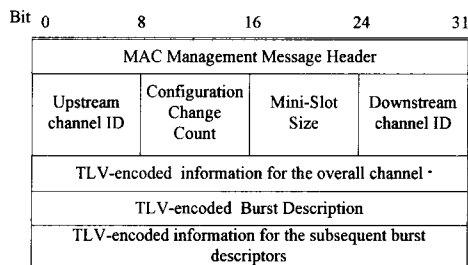


图 1 上行通道描述符的顶层结构

2.3 UCD 中用 TLV 编码的参数

在 UCD 中,除了配置改变计数器(Configuration Change Count)、微时隙大小(Mini-Slot Size)、上行通道 ID(Upstream channel ID)和下行通道 ID(Downstream channel ID),其它参数均以 TLV 数组形式编码。通道参数的类型值如表 1 所示。

表 1 通道 TLV 参数

名称	类型	长度	数值(长度可变)
Symbol Rate	1	1	基本速率(160 ksym/sec)的倍数(1, 2, 4, 8 或 16)
Frequency	2	4	上行中心频率(Hz)
Preamble Pattern	3	1~128	前同步码超级字符串
Burst Descriptor	4	n	可能不止一次出现

突发描述符(Burst Descriptor)采用的是混合 TLV 编码,包括基本型和结构型。这些编码是为每种类型的上行时间间隔用法而定义的,在该时间间隔中将使用物理层的参数。突发描述符的顶层编码格式如图 2 所示。

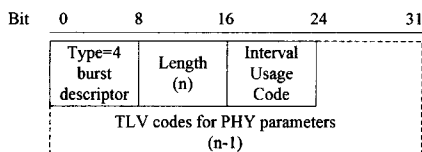


图 2 突发描述符的顶层编码格式

在每组突发描述符中将无顺序地列出了以 TLV 格式编码的物理层属性。这些参数的属性如表 2 所示。

表 2 上行物理层突发描述符中的参数

名称	类型	长度	数值
Modulation Type	1	1	1=QPSK; 2=16QAM
Differential Encoding	2	1	1=开; 2=关
Preamble Length	3	2	最大值 1024, 即 Preamble Pattern×8
Preamble Value Offset	4	2	标识出前同步码值的比特位
FEC Error Correction(T)	5	1	0~10
FEC Codeword Information Bytes(K)	6	1	固定: 16~253; 缩短: 16~253
Scrambler Seed	7	2	15bits
Maximum Burst Size	8	1	0~255, 表示在该类型突发期间传送的最大微时隙数
Guard Time Size	9	1	必须跟在该突发后的符号所占的时间
Last Codeword Length	10	1	1=固定的; 2=缩短的
Scrambler on / off	11	1	1=开; 2=关

图 3 给出了以 TLV 形式编码的 UCD 数据包内容部分的格式。

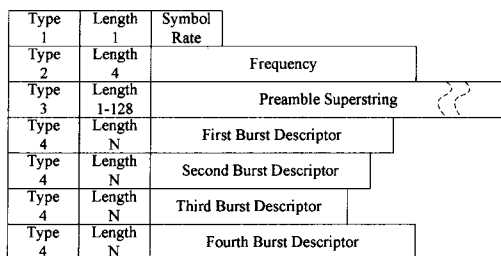


图 3 以 TLV 格式编码的 UCD

3 TLV 编码的特点及优化方法

3.1 TLV 编码的特点

通过在实践中使用 TLV 编码,以及对 TLV 编码特点的分析,本文认为这种编码方式的优点主要有以下 2 条:

- 1) 结构简单,含义清晰。
- 2) 编码灵活。主要表现在:参数根据需要可以出现或者不出现;参数出现的先后顺序不固定;参数数值的长度可以根据需要灵活变化。

由于 TLV 编码有上述的优点,所以在对它进行优化时,不仅要追求优化所带来的好处,还要尽量保留它原有的优点。

3.2 优化 TLV 编码的方法

图 4 是一组 UCD 上行物理层突发描述符的 TLV 编码实例,编码下方对应的说明是对数据包内容的解析。

上述实例中,TLV 编码的总长度是 339 字节。其中,用于存储长度的字段占到 63 字节,约是总长度的 18.58%。也就是说,如果去掉这些长度描述字段,这部分数据的长度能够缩短 18.58%。

从 DOCSIS 规范来说,规范中给出的参数表 1 和参数表 2 就是数据字典^[3]。在数据字典中,我们可以清楚地查找到存放任何参数所需要的空间大小,即默认的长度。例如,存放描述上行中心频率的数值所需要的空间是 4 个字节。然而,在整个网络协议系统中,实际情况可能多种多样,并非任何时候所有参数的数值长度都有确定的默认值可以遵循,所以当参数的数值长度不能确定是默认值时,必须在 TLV 编码中引入长度字段,以此来标识数值的长度,即对存储参数数

值的空间大小进行定义。本文提出的优化方法是将类型字段的最高比特位作为标志位,用来标识在此类型字段之后是否出现长度字段。

01 01 08 02 04 01 38 80 00
波特率 频率
03 80 CC CC F3 FF
前同步码模式(长度128字节)
04 25 01
时间间隔用法代码IUC=1
01 01 01 02 01 02
调制类型 差分编码
03 02 00 40 04 02 00 00
前同步码长度 前同步码的数值偏移
05 01 00 06 01 10
FEC纠错 FEC代码字信息字节
07 02 01 52 08 01 01
扰码因子 最大突发大小
09 01 10 0A 01 01
保护时间大小 最后代码字长度
0B 01 01
扰码器开关
04 25 03
时间间隔用法代码IUC=3(其后37字节TLV编码省略)
04 25 04
时间间隔用法代码IUC=4(其后37字节TLV编码省略)
04 25 05
时间间隔用法代码IUC=5(其后37字节TLV编码省略)
04 25 06
时间间隔用法代码IUC=6(其后37字节TLV编码省略)

图4 UCD的TLV编码数据

可以进一步规定:当标志位为1时,表示存在长度字段;当标志位为0时,表示不存在长度字段。当不存在长度字段时,数值的长度是数据字典中规定的该类型的默认长度。这种优化后的TLV结构如图5所示。

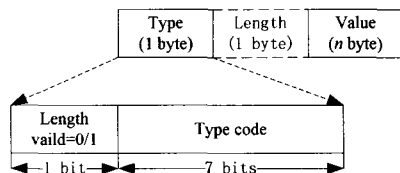


图5 优化后的TLV结构图

在实际编码和解码的过程中,长度字段是否出现就由Length valid位是0还是1来标识。这样优化的好处显而易见,可以有效缩短TLV编码的长度。

4 优化后的长度计算和性能分析

4.1 优化后编码的长度计算

针对UCD数据包,TLV优化之后,对数据包长度的计算方法是从小包原来的总长度中减去TLV结构中存储长度字段的字节个数。UCD中的TLV参数主要有3种形式,如图6所示。

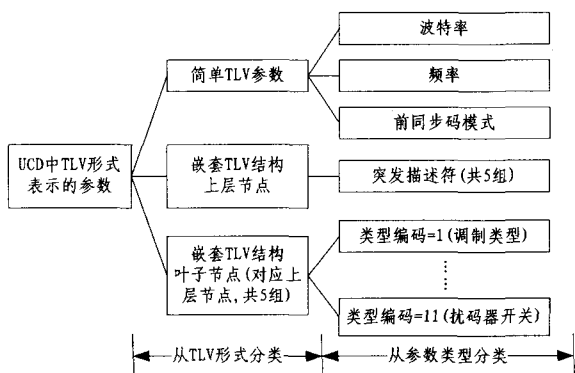


图6 UCD中TLV形式表示的参数

根据分类,优化后数据包长度计算的一般公式为

$$LengthNew = LengthTotal - C \cdot n_1 - C \cdot n_2 - \sum_{i=1}^{NumLeaf_i} C \cdot n_i \quad (1)$$

其中,C表示长度字段本身所占用的字节数, n_1 是简单TLV参数的个数, n_2 是嵌套TLV结构中包含的上层节点的个数, $NumLeaf_i$ 是嵌套TLV结构中第*i*个上层节点所对应的叶子节点的个数。

4.2 优化后的性能分析

以UCD中嵌套TLV结构的一组突发参数为例,对比优化前后TLV编码的长度。假设在这组参数中,使用默认长度的参数数目逐个增多,则编码长度的变化如图7所示。可以看出,数据包的长度明显缩短了。

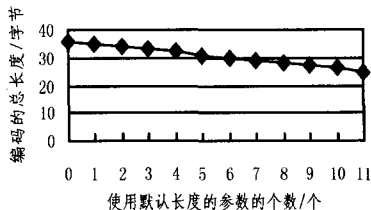


图7 优化一组突发参数

实验发现,CMTS周期性发送的UCD,其长度都是368字节(从MAC数据包头算起)。4种通道TLV参数每次都会出现,其中突发描述符类型包含5组突发参数。如果所有以TLV形式编码的参数都采用DOCSIS协议中规定的长度,那么在去掉长度字段之后,整个数据包的长度将是305字节,缩短了17.12%。相对于有限的网络带宽和传送时隙,数据包长度的缩短意味着传送这种数据包所需要的微时隙将缩短,从而网络传输的效率将大大提高。

本文提出的优化方法可以有效缩短TLV编码的长度,但其代价是增加了TLV编码的复杂性和缩短了类型编码的值域。

前一种代价可以从编码和解码两方面来分析。首先,可以确定编码方同解码方一样,熟知数据字典。在数据字典的规范下,绝大多数参数在绝大部分时间段内都将使用默认的数据长度进行参数编码。如果参数的长度不是默认值,或是在类型编码阶段不确定,则需要将类型字段的最高比特位设为1。软件编码的实现方法是对类型字段进行加0x80数学运算或与0x80进行逻辑或运算;硬件编码的实现方法是增加一个加法器或将最高比特位直接置1。解码的过程如果用软件来实现,是先进行判断,根据是否存在长度字段,选择不同的路径来解析数据。硬件实现解码的方法是与0x80做逻辑与运算或直接将类型的最高位引出,使其结果用于选择采用哪路部件进行下一步处理。可以发现,采用优化方法后,编码过程和解码过程需要一些改变,但无论软件系统还是硬件系统,增加的复杂度都不大。

为了研究优化方法缩短类型编码值域是否会带来新的问题,通过对一些网络协议的阅读,本文发现几乎所有网络协议规定的采用TLV编码的数据类型编码值都在127以内。因此,在很大范围内,本文提出的优化方法都能满足实际编码的需要。

结束语 抽象语法标记(ASN.1)是一种ISO/ITU-T标准,因此对TLV编码规则的使用遍及网络各层的诸多协议和规范。在设计网络通信系统的时候,如果能够对这种司空

(下转第261页)

表示操作标注。操作数量用于说明服务共包含几个操作,可取值为 $0, \dots, n$ (n 为操作的最大数量)。0表示该服务没有操作标注,当标注为服务能力标注或接口标注时,操作数量元素取值也为0。概念位置标识只在操作数量元素非零时有意义,具体表示该索引概念在某个具体服务中属于哪个操作。例如,概念 flightOrder 连接四元组 $\langle ss1, 1, 0, 0 \rangle$,表示服务 ss1 的服务能力标注包含概念 flightOrder;概念 orderFlight 连接四元组 $\langle ss1, 3, 2, 1 \rangle$,表示服务 ss1 有两个操作,其中第一个操作标注包含概念 orderFlight。

出于提高搜索速度的考虑,在实际设计时可将后3个概念索引表合并成一个索引表。索引表中索引项是一个三元组 \langle 服务标识,概念位置标识,标注类型 \rangle 。其中服务标识为服务唯一标识编号。概念位置标识用于区分概念属于服务描述中的哪个操作。标注类型用于匹配具体语义标注类型,可取值为1,2,3。其中1表示输入标注,2表示输出标注,3表示错误消息标注。例如概念 book 连接三元组 $\langle ss1, 1, 1 \rangle$,表示服务 ss1 的第一个操作的输入标注包含概念 book;概念 inventory 连接三元组 $\langle ss1, 2, 2 \rangle$,表示服务 ss1 的第二个操作的输出标注包含概念 inventory。服务资源语义索引库内部结构如图5所示。

4.4 断言及关键字索引

RDF4S 语义服务描述模型中 QoS 语义的服务级别 QoS 和操作级别 QoS 以及执行语义的前置操作和后置操作是通过断言进行声明的。因此,不同于有本体概念标注的语义,在资源匹配处理时必须基于断言进行匹配。与语义索引库的建立过程类似,同样建立倒排索引供资源匹配处理使用。

功能语义中的服务分类是通过分类法建立的。因此,对功能语义的服务分类进行资源匹配时可以只考虑传统的关键字匹配,而无需其它特别处理。对功能语义的服务分类涉及的关键字进行倒排索引后,建立服务分类关键字索引供资源匹配处理使用。

结束语 为使用户能快速、精确地搜索已发布的 RDF4S 语义服务资源,本文提出一种基于 RDF4S 语义服务描述模型的服务资源搜索框架,详细说明了该搜索框架的结构。整个基于 RDF4S 语义服务描述模型的搜索框架主要由资源获取

模块、RDF4S 语义服务资源快照、服务资源概念索引库、断言及关键字索引库、概念索引库和资源匹配模块6部分组成。RDF4S 语义服务资源的匹配主要分为断言匹配、关键字匹配和语义匹配。搜索框架根据 RDF4S 语义服务描述模型的 QoS、执行、功能和接口4种语义标注元素分别采用不同的搜索匹配策略,可以在很大程度上提高搜索的效率和精度。下一步的研究工作将围绕两方面进行:一方面将基于 Eclipse 开发平台实现上述基于 RDF4S 语义服务描述模型的服务资源搜索框架;另一方面将继续研究在资源匹配过程中不同匹配策略,以提高 RDF4S 语义服务资源搜索框架的搜索效率和精确度。

参考文献

- [1] Burstein M, Bussler C, Zaremba M, et al. A Semantic Web Services Architecture. *IEEE Internet Computing*, 2005, 9(5): 72-81
- [2] WSDL-S, W3C Member Submission on Web Service Semantics. <http://www.w3.org/Submission/WSDL-S/>
- [3] Martin D, Burstein M, Hobbs J, et al. OWL-S Semantic Markup for Web Services. W3C submission 2004. <http://www.w3.org/Submission/OWL-S/>
- [4] Roman D, Keller U, Lausen H, et al. Web Service Modeling Ontology. *Applied Ontology*, 2005, 1(1): 77-106
- [5] Lin L, Arpinar I B. Discovery of Semantic Relations between Web Services//*International Conference of Web Services*. 2006; 357-364
- [6] Paolucci M, Mawamura T, Payne T R, et al. Semantic Matching of Web Service Capabilities//*First International Semantic Web Conference 2002*. 2002; 333-347
- [7] Keller U, Lausen H, Stollberg M. On the Semantics of Functional Descriptions of Web Services//*3rd European Semantic Web Conference (ESWC)*. 2006; 605-619
- [8] 吴健, 吴朝晖, 李莹, 等. 基于本体论和词汇语义相似度的 Web 服务发现. *计算机学报*, 2005, 28(4): 595-602
- [9] 郭得科, 任彦, 陈洪辉, 等. 一种 QoS 有保障的 Web 服务分布式发现模型. *软件学报*, 2006, 17(11): 2324-2334

(上接第106页)

见惯的编码方式进行一些改进,在不破坏网络编码整体规范性的前提下,缩短数据包的长度无疑是提高通信效率的一种方法。本文提出的优化 TLV 编码的方法,虽然增加了一些编码和解码的复杂性,缩小了可以一次编码的参数类型的数量,但却可以有效地缩短网络上大量数据包的长度。随着处理器速度的提高,数据处理复杂性带来的影响正在减弱,而网络上与日俱增的数据量,以及有限的网络时间和带宽才是约束网络系统性能提高的瓶颈,所以本文提出的这种对 TLV 编码优化的方法有着广泛的应用前景。

参考文献

- [1] ITU-T Study Group. OSI networking and system aspects-Abstract Syntax Notation One, X. 690. 2002; 3-4
- [2] CableLabs. Data-Over-Cable Service Interface Specifications Radio Frequency Interface Specification. SP-RFIV1. 1-I06-001215. 2000; 82-85

- [3] SCTE. Data-Over-Cable Service Interface Specifications - Radio Frequency Interface Specification. 2002; 20-22
- [4] Fellows D, Jones D. DOCSIS cable modem technology[J]. *Communications Magazine*, IEEE, 2001, 39(3)
- [5] Sdralia V, Holcombe M. Ranging schemes for fast dynamic recovery of DOCSIS networks[J]//*Proc. 9th IEEE International Conference on Networks (ICON'01)*. Bangkok, Thailand, 2001
- [6] Sdralia V. Optimized recovery of DOCSIS networks using reserved persistent ranging[J]//*Proc. IEEE Globecom 2001*. San Antonio, TX, USA, 2001
- [7] Lin Y D, Yin W M, Huang C Y. An Investigation into HFC MAC Protocols[J]; *Mechanisms, Implementation, and Research Issues*. *IEEE Communication Surveys*. <http://www.comsoc.org/pubs/surveys>. 3rd Quarter 2000
- [8] Carroll M D. Aligning the Initial Maintenance Intervals of Cable Modem Upstream Channels[J]. *IEEE Communication Magazine*, September 2003